NASA/CR-97-

206407

IN-82-CR
5017

Final Technical Report for

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
Ames Research Center

A Toolkit for Active Object-Oriented Databases with Application to Interoperability

(Response to DARPA BAA 92-06)

NAG2-862

Roger King, Professor
Department of Computer Science
University of Colorado
Campus Box 430
Boulder, CO 80309-0430

OVERVIEW

In our original proposal we stated that our research would "develop a novel technology that provides a foundation for collaborative information processing." The essential ingredient of this technology is the notion of "deltas," which are first-class values representing collections of proposed updates to a database. The Heraclitus framework provides a variety of algebraic operators for building up, combining, inspecting, and comparing deltas. Deltas can be directly applied to the database to yield a new state, or used "hypothetically" in queries against the state that would arise if the delta were applied. The central point here is that the step of elevating deltas to "first-class" citizens in database programming languages will yield tremendous leverage on the problem of supporting updates in collaborative information processing. In short, our original intention was to develop the theoretical and practical foundation for a technology based on deltas in an object-oriented database context, develop a toolkit for active object-oriented databases, and apply this toward collaborative information processing.

INTRODUCTION TO H20/SST

It is generally desirable that systems that support human collaboration not interfere with user's preferred modes of working. The needs of such users include the ability to work on noninterfering versions of the artifact and the ability to employ a wide range of conflict resolution strategies. While such systems have data storage requirements, traditional database systems are poorly suited to meet those requirements due to their isolating concurrency policies and single state nature. To address this mismatch we have designed a nonlocking data storage manager, called H2O/SST, which provides functionality to support users in the process of resolving conflicts and inconsistencies resulting from their independent actions.

H2O/SST adds a State Specification Tree (SST) as an interface to the Heraclitus DBMS. A Heraclitus database directly transforms updates into delta values, which are syntactic representations of the update. Since the updates are the objects that are stored in the database, locking of the components of the artifact is avoided and users can work freely without interference from the actions of other users. The specification of the nature of the delta values is part of the application schema, allowing application specific definition of conflict. The SST adds the functionality to organize the delta values into meaningful states and provides the mechanisms to support conflict analysis and resolution. Importantly, the SST does not impose particular policies for conflict resolution that is left to the application. We illustrate the functionality of H2O/SST in the context of authoring systems.

DETAILED DESCRIPTION OF H2O/SST

It is recognized that traditional database management systems (DBMSs) are poorly suited for managing the artifacts shared by collaborating individuals. While systems that support cooperative work do have a need for persistent storage of the artifacts related to that work, it is often difficult to use a DBMS for that storage. The difficulty is due to fundamental differences between the requirements of systems to support collaborative work (groupware systems) and the requirements that most current DBMSs are designed to support.

One reason for the mismatch between database and groupware requirements is that traditional DBMSs are designed to maintain a single consistent database state which represents the information modeling some enterprise. Changes to the database state, which reflect changes to the enterprise, are generally made as in-place updates which effectively destroy previously held information. For this reason, traditional DBMSs do not provide easily used support for many of the activities required for collaborative endeavors, such as hypothetical actions and queries on the results of those actions, sharing of partially completed actions between cooperating individuals, and merging of individual actions toward the completion of a shared goal.

A second reason for the mismatch is that the usual database concurrency control protocols are specifically designed to isolate users and thus inherently hinder collaboration. Put simply, database protocols are designed to isolate competing users, while cooperative work requires protocols for coordinating collaborating users. Moreover, not only are the coordination protocols required by systems supporting cooperative work different from those traditionally supported by DBMSs, they are also difficult to define. Since the nature of shared work varies greatly across different groups, across different kinds of projects and over the lifecycle of a project, the coordination policies implemented by a groupware system must be flexible enough to adapt to such changing needs.

Recognizing this mismatch, the goal of the work presented here is to design a data storage manager better suited to supporting groupware requirements and to do it in such a way that it can be incorporated into existing DBMSs. The requirements on such a data storage manager are:

- It should allow for the existence of simultaneous, non-interfering versions of the artifact and provide flexible mechanisms for organizing updates into artifact versions.

- It should not impose constraining policies on how users can update the artifact or on how they resolve conflicts between their updates.

- It should provide facilities for detecting and analyzing conflicts between updates, and conflicts due to restructuring of the versions.

- It should retain concurrency control in a manner orthogonal to update creation.

To solve the problem stated above, we have designed H2O/SST that combines a Heraclitus (H2O) database with a State Specification Tree (SST). H2O/SST provides nonlocking updates, persistent state specification, and facilities to analyze conflicts arising between updates due to the lack of locking as well as conflicts due to manipulation of existing updates.
Heraclitus is an object-oriented DBMS that translates updates on the database state directly into delta values, thereby capturing state changes as manipulable objects. Since these delta values are the objects that are stored in the database, locking of application objects is bypassed, and concurrency control operates on the delta values themselves. Heraclitus also provides the mechanisms for computing states from the delta values and for combining delta values in various ways. Heraclitus is based on an algebra of deltas which provides a sound theory for reasoning about deltas and conflicts between deltas. Importantly, Heraclitus also allows the delta value operators to be customized as part of the application schema in such a way that the properties of the

underlying algebra of deltas are maintained, while allowing for application specific specification of the granularity and nature of conflict detection.

Heraclitus does not impose any restrictions on how delta values are organized to define states. The addition of the SST provides a simple mechanism for specifying states in meaningful ways, furthermore the SST is persistent and thus maintains state specifications as part of the database. In addition to being the mechanism for state specification, the SST provides facilities to support analysis of conflicts between updates, for merging updates and for resolving problems due to changing contexts when updates are moved within the SST.

H2O is built as on top of a commercial OODB (the current prototype runs on the Versant Object Database Management System, and operates as an extension to the ODMG-93 standard for OODBs. The H2O multistate manager interacts with the object manipulation interface to convert updates into delta values and to convert delta values into objects. The SST acts as an interface to the multistate manager of Heraclitus allowing controlling the delta values that the multistate manager uses to compute database states.

Since Heraclitus operates on delta values, rather than materialized states, and since it imposes no restrictions on how users manipulate those delta values, it leads to an environment in which conflict analysis and resolution are of prime importance. We've made substantial progress on the issue of defining and detecting conflict. The mechanism for this is the delta form which gives the schema designer considerable flexibility in designing defining conflict for specific data types.

Simply detecting conflict is only the first step of the problem, it must be followed by conflict analysis, to determine what is causing the conflict, and conflict resolution to determine what to do to solve the conflict. At the system level, Heraclitus only helps to detect conflict, being limited to failure when conflict is detected during update merges.

To deal with the question of conflict analysis and resolution, we looked to the domain of Computer Supported Cooperative Work (CSCW) which naturally deals with human strategies for dealing with conflict. Research in the field confirms that such strategies are extremely varied and variable. Furthermore, it is recognized that conflict is not an undesirable consequence of group work, but is in fact an integral part of it. With this in mind, the concrete goal of using Heraclitus to build a data storage manager that would be appropriate to support the needs of authoring systems was devised as a means of experimenting with the more abstract question of the role of a database system in conflict resolution. Recognizing the wide variety of approaches humans employ for dealing with conflict, we chose to limit the database's role to conflict analysis, but to never have it enforce specific policies. Such policies are left to the applications and/or users.
The H2O DBPL is not effective in approaching this problem because it lacks the notion of persistent state specification. The DBPL allows only the specification a single transient state from the available delta values. While this is sufficient for the support of hypothetical queries, it is not effective for general conflict analysis, because it is unaware of the contexts in which the updates are intended to be used. This observation lead to the development of the SST, which is a persistent representation of the states of interest. The SST thus replaces the H2O DBPL as the mechanism for interacting with the multistate management facility in Heraclitus. Since the SST is aware of the

user's perceptions of context, it can also analyze the update configurations that define the states and provide detailed information on the causes of conflict.

Example Application: Cooperative Authoring Systems

To demonstrate the applicability of this architecture in the context of groupware, we are designing and implementing a rudimentary authoring system prototype. However, because Heraclitus allows delta values and their operators to be customized for application defined data types, the results are not restricted to systems with text-like artifacts.

Since we are primarily interested in showing that H2O/SST can provide generic data storage services for authoring systems, rather than attempting to design new authoring systems, the prototype will be based on existing authoring system designs. The authoring system will support asynchronous editing and version management.

Besides acting as a test platform for H2O/SST, the authoring system will be unique in that it will support for flexible and redefinable collaboration policies, as well as different models for computing and analyzing updates on text objects can easily be supported.

PUBLICATIONS

R. Hull, R. King, G. Zhou, "Generating Data Integration Mediators that Use Materialization", Journal of Intelligent Information Systems, 1996.

M. Doherty and R. Hull and M. Rupawalla, "Structures for Manipulating Proposed Updates In Object-Oriented Database", June 1996, pp. 306-317, Proceedings of the ACM SIGMOD Conference on the Management of Data.

M. Derr and J. Durand and M. Doherty and R. Hull and M. Rupawalla, Applications of {Heraclitus} in Telecommunications Information Processing, International Journal of Engineering Intelligent Systems Special Issue on Databases and Telecommunications, July 1996.

M. Doherty and R. Hull and M. Derr and J. Durand, On Detecting Conflict Between Proposed Updates, September 1995, Proceedings of the International Workshop on Database Programming Languages, Springer-Verlag.

O. Boucelma and J.Dalrymple and M.Doherty and J. C. Franchitti and R.Hull and R. King and G. Zhou, Incorporating Active and Multi-database-state Services into an OSA-Compliant Interoperability Framework, The Collected Arcadia Papers, Second Edition, May, 1995, University of California, Irvine.